

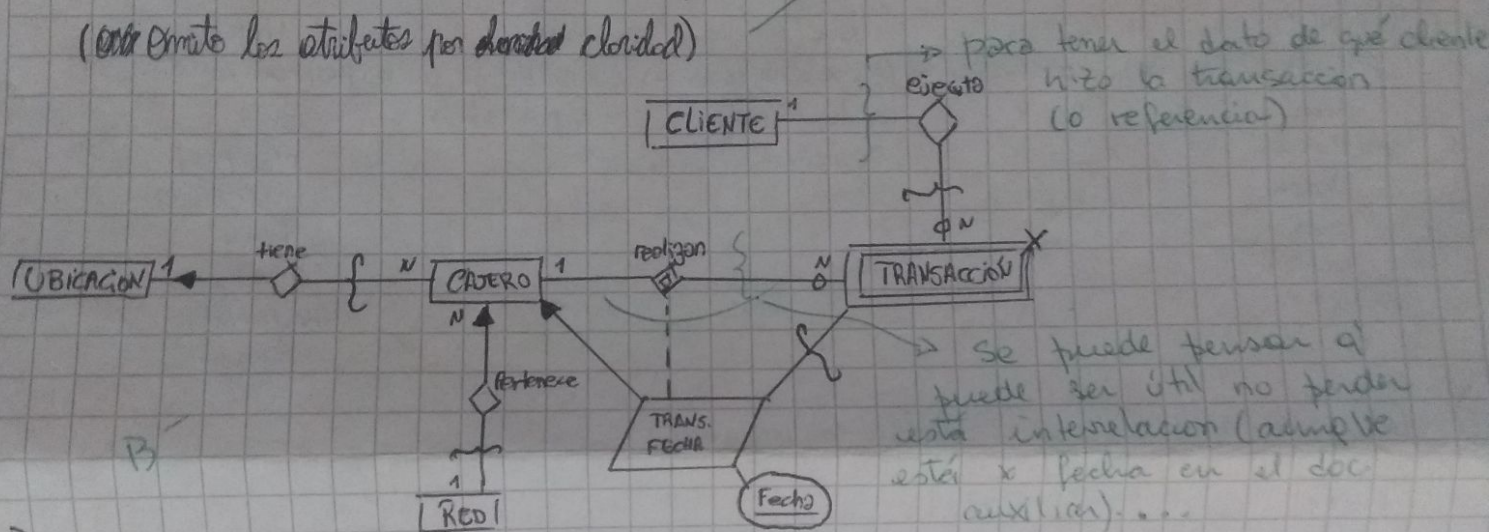
A  
Auditor

1. a) Como se desea buscar las transacciones por fecha y la base de datos puede ser muy grande, utilizo un documento auxiliar sobre la relación 'realizan', fortalezco por fecha. ✓

• Incrusto los cajeros a la ubicación, a la vez que incrusto la red al cajero, para poder acceder rápido a los datos. ✓

• El resto lo manejo con referencias, para no perder las relaciones entre las entidades. ✓

(no omito los atributos por dar claridad)



ii.

```
Ubicación: { "type": "object",
  "properties": {
    "idUbicación": { "type": "integer" },
    "Dirección": { "type": "string" },
    "Tipo": { "type": "string" },
    "Nombre": { "type": "string" },
    "Cajeros": { "type": "array",
      "items": { "type": "object",
        "properties": {
          "idCajero": { "type": "integer" },
          "Mareo": { "type": "string" },
          "Red": { "type": "object",
            "properties": {
              "idRed": { "type": "integer" },
              "Nombre": { "type": "string" }
            }
          }
        }
      }
    }
  }
}
```



b)

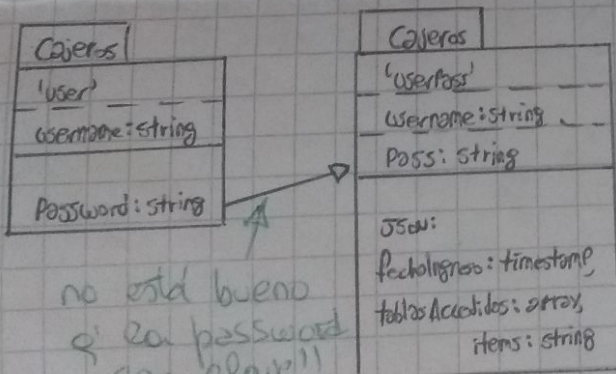
HR1	HR2	HR3	HR4	HR5
(i) TRANS x CAJERO x CLIENTE	TRANS x CAJERO x CLIENTE	TRANS x CAJERO x CLIENTE	(sin combite, ya que no se pide ordenar)	TRANS x CAJERO x CLIENTE
tiempo	Documento	id cliente K		id cliente K
fecha	id cliente K	Fecha CT		Fecha CT
importe	⋮	⋮		tiempo CT
id cajero	(quedo igual)	(quedo igual)		id cajero CT
Horario				importe
Nombre				Horario
Apellido				Nombre
Documento				Apellido
id cliente				Documento

• Asumo que se busca por id cliente, ya que pueden haber documentos repetidos.

(ii)

HR1	HR2	HR3	HR4	HR5
RED x CAJERO x UBIC.	RED x CAJERO x UBIC.	(sin combite, ya que no hay búsqueda por desigualdad)	(sin combite, ya que no se pide ordenar)	RED x CAJERO x UBIC.
id Red	id Red K			id Red K
red. Nombre	⋮			id cajero CT
id cajero	(quedo igual)			id Ubicación CT
Horario				red. Nombre
id Ubicación				Horario
ubic. Nombre				ubic. Nombre
Dirección				Dirección

c) • Asumo que a las tablas se las identifica con un nombre.



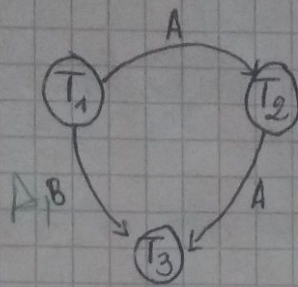
no está bueno  
q' el password  
sea clave!!  
(hubieras usado  
con idUser)



2/2

2.

(i)



- Es serializable porque no hay ciclos.
- El único orden serial se desprende de orden topológico del grafo:  $T_1, T_2, T_3$

(ii) •  $T_2$  lee de  $T_1$

•  $T_3$  lee de  $T_1$  y  $T_2$

• Como  $T_2$  lee de  $T_1$  antes que  $T_1$  comita la historia no es ACK (y por los timestamps no es estrueto).

• Como el orden de los commits es  $c_1, c_2, c_3$ , la historia es Recuperable

(iii)

•  $T_1$  no es 2PL porque toma otro lock después de liberar uno.

•  $T_2$  es 2PL, pero no es 2PLE porque libera el lock de escritura antes de comitar.

•  $T_3$  es análogo a  $T_2$ .

b)

(i)

$T_1 = 100$	$T_2 = 200$	$T_3 = 300$
$r(A)$ $RT(A) = 100$ $C(A) = T$		
$C$		
		$r(C)$ $RT(C) = 300$ $C(C) = T \text{ no}$
	$r(C)$ $RT(C) = 300$ $C(C) = T$	
		$w(A)$ $WT(A) = 300$ $C(A) = F$
	$r(B)$ $RT(B) = 200$ $C(B) = T$	
		$r(B)$ $RT(B) = 300$ $C(B) = T$
	$w(B)$	$C$

→ no hay problema porque  $T_3$  es más reciente

• (write-too-late) Yo que  $TS(T_2) < RT(B)$ , y  $T_3$  no aborta.  
Se aborta  $T_2$  y se reinicia con otro timestamp

• Si se invierte el orden de  $T_2$  y  $T_3$ , no ocurre el write-too-late (dado que  $TS(T_2) = RT(B)$ ) y la operación  $w_2(A)$  también se puede ejecutar sin problemas porque  $T_3$  ya comitó y  $T_2$  es posterior a ella.



(ii) Los planificaciónes multiversion permiten resolver el problema de read-too-late, pero como ocurre un write-too-late (en el primer caso), no cambia nada. ✓

Como como en el segundo caso no se presentan problemas, tampoco hace diferencia. ✓  
(Se generan las nuevas versiones con el write)

c)

(i)

(A) Se deben deshacer las operaciones de  $T_3$  y  $T_4$ , agregando  $\langle \text{ABORT } T_3 \rangle$  y  $\langle \text{ABORT } T_4 \rangle$  al

final del log. Los valores quedan como:

X	Y	Z
6	90	7

 ✓

(B) No, no podría operar porque  $T_3$  no comités todavía. ✓

(ii)

(A) Se deben rechazar las operaciones de  $T_4$  y  $T_3$ , agregando  $\langle \text{ABORT } T_3 \rangle$  y  $\langle \text{ABORT } T_4 \rangle$  al final

del log. Los valores quedan como:

X	Y	Z
41	3	17

 ✓

(B) Sí, podría operar en caso de que los cambios de  $T_2$  ya hayan sido hechos a disco. ✓